

1. Drawing filed on 6/26/03 is accepted by the examiner.

***EXAMINER'S AMENDMENT***

2. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

3. Authorization for this examiner's amendment was given in a telephone interview with Mr. David Sakata, Reg. #59,949 on 7/28/09.

4. Amending the specification as follow:

- a. Amendments to the Title:

OPERATING SYSTEM MANAGING A LINKED LIST OF CALLBACK  
DYNAMIC FUNCTION TABLES FOR ACQUIRING EXCEPTION HANDLING  
INFORMATION FROM A RUNTIME ENVIRONMENT

- b. Amend page 25 lines 22-25 of the specification to read:

Computer 1002 typically includes a variety of electronically-accessible media. Such media may be any available media that is accessible by computer 1002 or another electronic device, and it includes both volatile and non-volatile media, removable and non-removable media, and storage ~~and transmission~~ media.

5. Amending the claims as follow:

1.-5. (Canceled)

6. (Currently Amended) One or more electronically-accessible media ~~comprising~~ having stored thereon electronically-executable instructions that utilize an application programming interface, the application programming interface facilitating creation of callback-type dynamic function tables; each callback-type dynamic function table including a begin address, an end address, and a callback function, each callback-type dynamic function table corresponding to a code heap that stores code for a plurality of functions in a runtime environment; wherein interaction between the runtime environment and an operating system is precipitated upon calling the callback function to acquire exception handling ~~and~~ or unwind information;

wherein the interaction between the runtime environment and the operating system comprises (i) the operating system requesting exception handling or unwind information from the runtime environment and (ii) the

runtime environment providing the exception handling or unwind information to the operating system;

wherein the begin address and the end address of each callback-type dynamic function table comprises a begin address and an end address of a respective code heap to which a respective callback-type dynamic function table corresponds;

wherein the exception handling or unwind information comprises data for code address-to-pointer information for a particular function of the plurality of functions; and

wherein each callback-type dynamic function table is created, at least in part, by adding each callback-type dynamic function table to a linked list of dynamic function tables.

7.-9. (Canceled)

10. (Currently Amended) The one or more electronically-accessible media as recited in claim [[9]] 6, wherein the particular function of the plurality of functions is ascertained using an instruction pointer that is passed as an argument of the callback function.

11. (Original) The one or more electronically-accessible media as recited in claim 6, wherein each callback-type dynamic function table is created, at least in part, by adding each callback-type dynamic function table to a linked list of dynamic function tables.

12. (Original) The one or more electronically-accessible media as recited in claim 11, wherein the linked list of dynamic function tables includes one or more non-callback-type dynamic function tables.

13.-16. (Canceled)

17. (Currently Amended) An electronic device comprising:  
at least one processor;  
a system memory;  
a runtime environment that is managing code for a plurality of functions;  
and  
an operating system that is managing a linked list of dynamic function tables ~~that are~~ including a callback dynamic function table, wherein the linked list is searched when an exception occurs, wherein the callback dynamic function table comprises a begin address, an end address and a callback function, each callback dynamic function table corresponding to a code heap that stores code for the plurality of functions;

the operating system ~~adapted~~ configured to call ~~[[a]]~~ the callback function when the exception occurs within an address range bounded by the begin address and the end address of as indicated by the callback ~~[[a]]~~ dynamic function table of the linked list of dynamic function tables to request that the runtime environment provide exception handling ~~and/or~~ unwind information for at least one function of the plurality of functions;

wherein the runtime environment is ~~capable of providing~~ configured to provide to the operating system the exception handling ~~and/or~~ unwind information for the at least one function of the plurality of functions responsive to calling the callback function.

18.-21. (Canceled)

22. (Original) The electronic device as recited in claim 17, wherein the runtime environment comprises at least one just-in-time (JIT) compiler.

23. (Currently Amended) The electronic device as recited in claim 17, wherein the exception handling ~~and/or~~ unwind information comprises data for code address-to-pointer information for the at least one function of the plurality of functions.

24. (Original) The electronic device as recited in claim 23, wherein the data for code address-to-pointer information for the at least one function of the plurality of functions directly or indirectly provides to the operating system a start address, a final address, and an unwind pointer for the at least one function of the plurality of functions.

25.-39. (Canceled)

40. (Currently Amended) One or more electronically-accessible media ~~comprising~~ having stored thereon electronically-executable instructions that include:

a callback function, the callback function accepting as input an instruction pointer that is associated with an address of a function from a runtime environment and producing as output data for code address-to-pointer information for the function having the address that is associated with the instruction pointer;

wherein the callback function may be called by an operating system and implemented by the runtime environment; and

the operating system managing a linked list of dynamic function tables including a callback dynamic function table, wherein the linked list is searched when an exception occurs, wherein the callback dynamic function table comprises a begin address, an end address and the callback function;

the operating system configured to call the callback function when the exception occurs within an address range bounded by the begin address and the end address of the callback dynamic function table of the linked list of dynamic function tables to request that the runtime environment provide exception handling or unwind information for the function.

41. (Canceled)

42. (Currently Amended) The one or more electronically-accessible media as recited in claim ~~[[41]]~~ 40, wherein the ~~callback dynamic function table includes a~~ begin address and ~~[[an]]~~ the end address ~~that~~ define an address range that includes the instruction pointer.

43. (Original) The one or more electronically-accessible media as recited in claim 40, wherein the data for code address-to-pointer information that is output by the callback function is provided from the runtime environment to the operating system.

44. (Original) The one or more electronically-accessible media as recited in claim 40, wherein the data for code address-to-pointer information comprises a reference to the code address-to-pointer information.

45. (Original) The one or more electronically-accessible media as recited in claim 44, wherein the referenced code address-to-pointer information comprises a start address, a final address, and an unwind pointer for the function having the address that is associated with the instruction pointer.

46. (Original) The one or more electronically-accessible media as recited in claim 40, wherein the data for code address-to-pointer information comprises the code address-to-pointer information.

47. (Canceled)

48. (Original) The one or more electronically-accessible media as recited in claim 40, wherein the callback function is implemented, at least partially, by the runtime environment by inspecting a code header for the function having the address that is associated with the instruction pointer.

49. (Original) The one or more electronically-accessible media as recited in claim 48, wherein the data for code address-to-pointer information is derived using the code header.



50. (Original) The one or more electronically-accessible media as recited in claim 48, wherein the code header is ascertained using a heap structure contents of a code heap that includes code for the function having the address that is associated with the instruction pointer.

51.-53. (Canceled)

54. (Currently Amended) The one or more electronically-accessible media as recited in claim ~~[[53]]~~ 40, wherein the callback dynamic function table having the callback function corresponds to at least two functions compiled in the runtime environment by a just-in-time (JIT) compiler.

55. - 64. (Canceled)

65. (Currently Amended) An electronic device comprising:

at least one processor; and

one or more media ~~including storing~~ processor-executable instructions ~~that are capable of being executed by the at least one processor, the processor-executable instructions adapted to cause~~ when executed by the at least one processor causing the electronic device to perform actions comprising:

in response to detecting an exception, searching a linked list of dynamic function tables using an instruction pointer, wherein the dynamic function tables include a callback dynamic function table, wherein the callback dynamic function table comprises a begin address, an end address and a callback function, each callback dynamic function table corresponding to a code heap that stores code for a plurality of functions managed by a runtime environment;

locating [[a]] the callback dynamic function table having [[a]] the callback function; wherein the locating action comprises:

locating the callback dynamic function table having the callback function from the dynamic function table linked list using the instruction pointer, the callback dynamic function table including a begin address that is less than and an end address that is greater than the instruction pointer;

initiating, by an operating system, the callback function to interact with [[a]] the runtime environment;

receiving data for code address-to-pointer information from the runtime responsive to the initiating; and

using the received data for code address-to-pointer information to attain information for ~~at least one of~~ exception handling ~~and/or~~ stack unwinding.

66. (Original) The electronic device as recited in claim 65, wherein:  
the receiving action comprises:

receiving a reference to code address-to-pointer information from the runtime; and

the using action comprises:

using the reference to attain code address-to-pointer information,  
including an unwind pointer; and  
using the unwind pointer to attain unwind information.

67. (Original) The electronic device as recited in claim 65, wherein the processor-executable instructions are adapted to cause the electronic device to perform a further action comprising:

discovering an exception from a runtime function.

68. (Canceled)

69. (Currently Amended) The electronic device as recited in claim ~~[[68]]~~65, wherein the processor-executable instructions are adapted to cause the electronic device to perform further actions comprising:

receiving at the runtime, responsive to the initiating, the instruction pointer with a request for the data for code address-to-pointer information;

ascertaining a runtime function associated with the instruction pointer;

accessing a code header of the ascertained runtime function to extract the data for code address-to-pointer information for the ascertained runtime function; and

providing, from the runtime to the operating system, the data for code address-to-pointer information.

70. -72. (Canceled)

73. (New) One or more electronically-accessible media having stored thereon electronically-executable instructions to perform acts comprising:

managing code via a runtime environment for a plurality of functions;

managing via an operating system a linked list of dynamic function tables including a callback dynamic function table, wherein the list is searched when an exception occurs, wherein the callback dynamic function table comprises a begin address, an end address and a callback function, each callback dynamic function table corresponding to a code heap that stores code for the plurality of functions;

calling via the operating system the callback function when the exception occurs within an address range bounded by the begin address and the end address of the callback dynamic function table of the linked list of dynamic function tables to request that the runtime environment provide exception handling or unwind information for at least one function of the plurality of functions;

wherein the runtime environment is configured to provide to the operating system the exception handling or unwind information for the at least one function of the plurality of functions responsive to calling the callback function.

74. (New) The one or more electronically-accessible media as recited in claim 73, wherein the runtime environment comprises at least one just-in-time (JIT) compiler.

75. (New) The one or more electronically-accessible media as recited in claim 73, wherein the exception handling or unwind information comprises data for code address-to-pointer information for the at least one function of the plurality of functions.

76. (New) The one or more electronically-accessible media as recited in claim 73, wherein the data for code address-to-pointer information for the at least one function of the plurality of functions directly or indirectly provides to the operating system a start address, a final address, and an unwind pointer for the at least one function of the plurality of functions.

***Reasons for Allowance***

6. The following is an examiner's statement of reasons for allowance:

Art Unit: 2194

7. The prior arts of record do not expressly teach or render obvious, in the context of the claims taken as a whole, the invention as recited in independent claims 6, 17, 40, 65 and 73.

8. Employing a stack to store information for multiple functions which includes information useful when an exception occurs, and unwinding of the stack to retrieve the information pertaining to a specific function when an exception occurred to properly handle the exception were disclosed substantially in US Patents 7,426,719, 6,293,712, 7,007,198 and 6,014,515 and US PG Publications 20030023905, 20030079202, 20040128658 and 20020188931. Employing an unwind table to unwind various information for exception handling was disclosed by US Patents 7,426,719, 6,293,712 and 6,014,515. The references do not expressly teach or render obvious the linked list of callback dynamic function tables managed and used by an operating system to acquire exception handling or unwind information from a runtime environment taken as a whole as recited in independent claims 6, 17, 40, 65 and 73.

9. Neither a reference uncovered that would have provided a basis of evidence for asserting a motivation, nor one of ordinary skilled in the art at the time the invention was made, knowing the teaching of the prior arts of record would have combined them to arrived at the present invention as recited in the context of independent claims 6, 17, 40, 65 and 73 as a whole.

10. Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue

Art Unit: 2194

fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

11. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Qing-Yuan Wu whose telephone number is (571)272-3776. The examiner can normally be reached on 8:30am-6:00pm Monday-Thursday and alternate Friday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Hyung (Sam) Sough can be reached on (571) 272-6799. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Hyung S. Sough/  
Supervisory Patent Examiner, Art Unit 2194  
08/03/09

/Qing-Yuan Wu/ Examiner, Art Unit  
2194